

REMARKS

Applicants appreciate the continued examination provided by the Examiner in the above-referenced application. Applicants have amended the independent claims to further clarify that the communication between, for example, the monitor application and the server follows an HTTP request-response communications model. As discussed in greater detail below, the cited references, either singularly or in combination, do not disclose or suggest this type of communications model between a client and a server for providing updated legacy host screen information to the client. Applicants have also cancelled Claims 8-18 without prejudice or disclaimer.

For the sake of brevity, Applicants' comments herein focus, for the most part, on the amendments to the independent claims. However, in order to ensure that the present Amendment After Final is fully responsive to the final Official Action, Applicants hereby incorporate all of Applicants' previous responses herein by reference. Furthermore, Applicants submit that the dependent claims are patentable at least as depending from a patentable base claim.

Applicants respectfully request entry of the present Amendment After Final as it places the case in condition for allowance while raising no new issues. For example, the claims, as previously written, recited in the preamble that the client application utilized a request – response communications model. The amendments, therefore, further clarify what was already present in the claims: that the request – response communications model is according to the HTTP standard. Alternatively, Applicants request entry of the present Amendment After Final as narrowing the issues for further prosecution and/or appeal.

The amended independent claims are patentable over Nakabayashi in view of Butts.

Claims 1 – 36 stand rejected under 35 U.S.C. § 103 over U.S. Patent No. 5,905,866 to Nakabayashi et al. ("Nakabayashi") in view of U.S. Patent No. 5,754,830 to Butts et al. ("Butts"). *Final Official Action, page 5*. The independent claims have been amended to further clarify that communication between the client (*i.e.*, the monitor application or notification code) and the server operates according to an HTTP response-

request communications model. For example, independent Claim 1 has been amended to recite in part:

establishing a first connection between the client application and a server application, wherein the server application provides updated legacy host screen information to the client application in response to requests from the client application **using an HTTP request-response communications model**, wherein the updated legacy host screen information is based on information formatted for a character terminal of a host legacy system;

establishing a second connection between a monitor application and the server application;

receiving a notification of the availability of updated legacy host screen information via the second connection at the monitor application **using the HTTP request-response communications model**;

requesting the updated legacy host screen information over the first connection **using HTTP request-response communications model** responsive to receiving the notification;

receiving the requested updated legacy host screen information at the client application; and

displaying the received updated host screen information utilizing the client application.

Independent Claims 19, 23, 30 and 34 include similar recitations. As discussed in Applicants' previous responses, Nakabayashi relates to the management of updated web pages. For example, Nakabayashi, column 44, lines 1 – 44, discusses the management of hypertext data which is associated with web pages, not information formatted for a character terminal of a legacy host system as recited in the independent claims. Accordingly, Nakabayashi does not relate to terminal emulation. Moreover, the fact that the system in Nakabayashi deals only with web pages illustrates that Nakabayashi does not need to deal with the type of asynchronous problem addressed by the present invention. In particular, because Nakabayashi deals with updating data from web pages, the communications used by Nakabayashi's system is synchronous in nature (*i.e.*, request-response). In contrast, embodiments according to the present invention, deal with synchronizing the asynchronous communications of terminal emulation using web type communications (*i.e.*, HTTP). For example, as discussed in the background section of the present application:

Some applications that run on legacy host systems can be accessed using a display terminal running a terminal protocol. The terminal protocol may enable communications to and from the display terminal, such as when a screen is transmitted to the display terminal and when user input is transmitted to the host system. Such protocols are sometimes referred to as "2 way synchronous" communications. In such a terminal protocol, for example, updated (or new) screens generated by the host application may be transmitted to the display terminal without a request from the user. In other words, updated screens may be automatically transmitted to the display terminal...

Unfortunately, some of the communications protocols used to provide terminal emulation between browsers and legacy host systems may not provide the same communications functions provided by the terminal protocols described above. For example, the Hypertext Transport Protocol (HTTP) utilizes a synchronous "request-response communications model." In HTTP, the server typically only provided information to the browser in response to a request from the browser. In such a system, it may be difficult to provide the asynchronous communications described above. In particular, it may be difficult to provide updated screens to the browser automatically. *Application, page 1, line 5 to page 2, line 7.*

As demonstrated by the above-cited passage from the background of the present application, the type of protocol used to transport web pages, (*i.e.* HTTP) is not, by itself, well suited to deal with the asynchronous type of communications used for terminal emulation. Accordingly, the system discussed in Nakabayashi does not appear to be appropriate to deal with asynchronous communications, such as terminal emulation.

In contrast to Nakabayashi, embodiments according to the present invention address the asynchronous problem outlined above by implementing notification code associated with the client. In some embodiments according to the invention, the notification code receives notification of the availability of updated legacy host screen information from the server according to an HTTP response-communications model. The notification code then requests that the server send the updated legacy host screen information to the client using the HTTP request-response communications model. Embodiments according to the present invention can, therefore, provide a way to synchronize the updated information generated by the host asynchronously. Accordingly,

Nakabayashi does not disclose or suggest the use of an HTTP request-response communications model for receiving notification of updated host information and requesting that the updated host screen information be transmitted as recited in the amended independent claims.

As understood by Applicants, Butts also does not disclose or suggest the use of an HTTP request-response communications model for receiving notification of updated host information and requesting that the updated host screen information be transmitted. In particular, the system in Butts is described as using a persistent TCP/IP socket connection to communicate with the server. For example, as shown in Figure 1 of Butts, the output process 42 communicates with the client thread 28 over a persistent TCP/IP socket connection 44. A persistent TCP/IP socket connection does not disclose an HTTP response-request communications model. In fact, as understood by Applicants, persistent TCP/IP socket connections actually avoid the asynchronous communications problems discussed above in reference to HTTP.

As understood by those skilled in the art, communicating using HTTP can introduce inefficiencies into the communication by, for example, incurring overhead associated with opening and closing a connection for each exchange that takes place between a client and a server. For example, according to HTTP request-response communications model, for data to be transmitted between the server and the client, a connection is opened between the server and the client upon a request for data. Once a response to the request is issued (*i.e.*, the data is provided) the connection between the client and server is closed. Accordingly, a new connection typically needs to be established for any subsequent communications. In contrast to the HTTP request-response communications model, a persistent TCP/IP socket connection is meant to avoid closing the connection such that new connections need not be established for further communications. **In other words, use of persistent TCP/IP socket connections is a way to avoid the overhead of using HTTP.** Accordingly, persistent TCP/IP socket connections do not disclose or suggest the claimed HTTP request-response communications model used in to provide terminal emulation.

Accordingly, even if Nakabayashi and Butts were to be combined, the combination would not disclose or suggest at least the HTTP request-response

communications model used for requesting and receiving the updated host screen information as recited in the amended independent claims.

Furthermore, there is no clear and particular evidence of a motivation of suggestion to combine Nakabayashi and Butts as required under § 103. As discussed in Applicants' previous responses, Nakabayashi does not discuss legacy host systems or providing terminal emulation for those systems, whereas Butts appears to relate to terminal emulation. Accordingly, there is no clear and particular evidence of a motivation or suggestion to combine these references as they appear to discuss different subjects and appear to solve completely different problems.

Furthermore, the evidence cited by the final Official Action as evidence of a motivation to combine Nakabayashi and Butts does not meet the requirement that the evidence be clear and particular. For example, the final Official Action states that the motivation or the combination is provided by Butts, column 1, lines 12 – 40 which states:

Many organizations operate computer network environments that include legacy host systems which store data and provide applications important to the operation of the organization. Such legacy host systems can include IBM mainframes (MVS, VM and VSE environments), IBM AS/400 systems and UNIX host systems.

It is desirable for such organizations to provide connection to the legacy host systems through terminal sessions on distributed client systems such as personal computers and computer workstations. This connection to the legacy host system provides access for users of the client systems to the data and applications on the legacy host system. These terminal sessions can include 3270, 5250, NVT and VT220 type terminal sessions.

One conventional method for providing terminal sessions is to execute a terminal emulator application on the client systems that connects directly to a host legacy system using a TCP/IP socket connection. Another conventional method is to provide connection through a web browser application by translating standard legacy data flows into HTML pages. However, such conventional web browser methods suffer from an inability to handle real-time host updates to user screens as well as other significant problems. For example, forms-based HTML/TN3270 packages are unable to overcome a range of problems associated with common HTML implementations such as real-time host

updates to user screens or finding a user's browser platform address on the network.

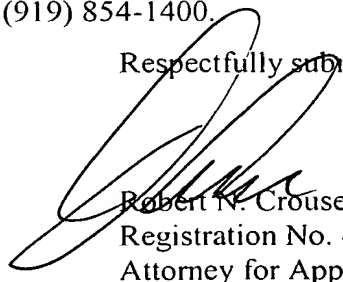
As demonstrated by the above cited passage of Butts, Butts focuses on terminal emulation whereas, Nakabayashi relates to the management of updated web pages and does not appear to have anything to do with terminal emulation. Applicants respectfully submit that the passage cited by the final Official Action does not provide the clear and particular evidence of a motivation or suggestion to combine references as required under § 103 by the case law discussed in Applicants' previous responses.

Applicants respectfully submit that amended independent claims 1, 19, 23, 30 and 34 are patentable over Nakabayashi and Butts for at least the reasons discussed above. Furthermore, the dependent claims are patentable at least per the patentability of the amended independent claims.

CONCLUSION

Applicants respectfully request entry of the present Amendment After Final as no new issues are raised by the present amendment. Applicants have shown that Nakabayashi and Butts, either singularly or in combination, do not disclose or suggest terminal emulation for host legacy system provided by an HTTP request-response communications model as recited in the amended independent claims. Applicants have further shown that there is no clear and particular evidence of a motivation or suggestion to combine Nakabayashi and Butts. Accordingly, Applicants respectfully submit that all claims are patentable and request the withdrawal of all rejections and the allowance of all claims. If any informal matters arise, the Examiner is encouraged to contact the undersigned by telephone at (919) 854-1400.

Respectfully submitted,



Robert N. Crouse
Registration No. 44,635
Attorney for Applicants

Correspondence Address:

In re: Webb et al.
Serial No.: 09/394,536
Filed: September 10, 1999
Page 15



20792

PATENT TRADEMARK OFFICE

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: MS AF, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on May 22, 2003.

Audra Wooten

Date of Signature: May 22, 2003